The third machine learning project

Your work on this project is to be presented to the rest of the class on Wednesday, April 29th. In the project, you can either choose to employ automatic programming or alternatively at least three methods from the scikit-learn, R or Weka toolboxes that are different from the ones used in previous projects. We will first present the automatic programming option and then the toolbox alternative.

1 The automatic programming option

- Give a small introduction to artificial evolution and automatic programming.
- Use your experiences from the previous two projects to determine if you should choose the same data set as before or a new one.
- Describe how you have made a specification file for ADATE and what it contains. How is your data coded? List the relevant user call-backs and give a description of each one.
- Discuss relevant parts of the log, trace and validation files for one of your runs. Give an analysis of the overfitting by comparing the performance on training data with the performance on validation data.
- Find the syntactic fingerprint for the program with the best validation value which is given by the last occurrence of "best validated" in the trace file. Locate this program in the log file and describe how it works.
- Use the script `ancestors.pl` to extract the genealogical chain for the best program. Give an overview of the chain including some especially interesting evolutionary steps.
- Choose a compound transformation from the chain or from the log file and list all intermediary programs generated by the compound transformation. Try to include at least one example of each program transformation.

- Look at the validation file and describe the relationship between run time and evaluation value. Do you think that more run time than you were able to use would give better results?
- Do classifications have differing costs? How can you enable ADATE to take this into consideration?
- Evaluate how good results that you have obtained. It is more important to give a correct evaluation and work systematically than to obtain the best possible performance.
- What differences and similarities do you see between artificial evolution in an ADATE run and natural evolution? In what ways is natural evolution superior? What advantages does ADATE have compared with natural evolution? What limits are there for what ADATE can accomplish?
- Criticize and compare ADATE with neural nets and C5.0 according to a number of suitable criteria that you choose yourself.
- What suggestions do you have for improvement of ADATE and associated tools? Use your experiences from the project to describe what should be changed or added to ADATE.
- What future improvements are there?

2 The toolbox alternative

If you choose this alternative, you are expected to learn your selected toolbox on your own since it is not taught in the lectures. Some popular machine learning toolboxes that you can choose are scikit-learn, R and Weka.

The scikit-learn toolbox is popular in Kaggle competitions, but may require that you know a little Python. It has been growing fast in popularity the last two years.

R is a statistics package that also contains machine learning algorithms. It is used in the Microsoft Azure Machine Learning toolbox, but otherwise available as open source software. It requires you to learn a bit of the R language.

Weka is the easiest to use, but does not work well if you have very many attributes or big data sets.

Proceed as follows.

- Read the user manual to get an overview of all the different machine learning algorithms that are available.

- Select at least three of these algorithms, for example one Bayesian method, one nearest neighbour algorithm and one support vector machine algorithm.
- Find relevant literature about each method that you have chosen and write a thorough description of each one including its specific implementation in the toolbox.
- Import your dataset into the toolbox.
- Run each algorithm with various combinations of options including cross-validaton.
- Compare the results from the three different algorithms with each other and with the ones from C5.0 / Cubist and neural nets in Matlab, Torch or Pylearn.
- Look at all the points in the project descriptions for the first and the second projects and include the ones that are relevant for your chosen methods.